## Ruby master - Feature #7006

Add Enumerable#grouped_in as a each_slice without the block

09/12/2012 03:34 AM - rosenfeld (Rodrigo Rosenfeld Rosas)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Rejected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
</tr>
<tr>
<td>Target version</td>
<td>2.0.0</td>
</tr>
</tbody>
</table>

### Description

I'd like to transform `[1, 2, 3, 4]` into `1, 2], [3, 4].`

Currently I can use something like:

```ruby
a = []; [1, 2, 3, 4].each_slice(2){|chunk| a << chunk}
```

But I'd prefer to write just:

```ruby
a = [1, 2, 3, 4].grouped_in(2)
```

Or any other method name you would choose, like "slices" or something else.

### History

#### #1 - 09/12/2012 03:48 AM - jeremyevans0 (Jeremy Evans)

This should work:

```ruby
a = [1, 2, 3, 4].each_slice(2).to_a
```

I don't see a reason to create another method to do this.

#### #2 - 09/12/2012 04:24 AM - marcandre (Marc-Andre Lafortune)

- Status changed from Open to Rejected

Indeed, `to_a` is the answer. Note that ActiveSupport has a similar `in_groups_of` that will return the results even when no block is given.

#### #3 - 09/12/2012 08:11 PM - rosenfeld (Rodrigo Rosenfeld Rosas)

Thanks, I didn't know that. I can't close this issue myself. Could someone do that, please?

#### #4 - 09/12/2012 11:53 PM - marcandre (Marc-Andre Lafortune)

rosenfeld (Rodrigo Rosenfeld Rosas) wrote:

> Could someone do that, please?

Already done.

#### #5 - 09/13/2012 07:28 AM - rosenfeld (Rodrigo Rosenfeld Rosas)

Ah, okay, thank you, didn't notice before :P